

Claims

1. A system for implementing a modular software system in a distributed computing environment, the distributed computing environment having a plurality of networked processing resources, comprising:

a first and a second general runtime component which are capable of communicating with each other;

the first and second general runtime components each comprising a plurality of sockets for connecting to a lower level component and managing interactions between the lower level components;

the first and second general runtime components each capable of operating on one of the plurality of networked processing resources;

a first and a second container component for engaging one of the plurality of sockets of one of the general runtime components;

the first container component implements a first predetermined functionality and comprises a first container coordinator for allowing the second container component to communicate with the first container component through their respective general runtime components; and

a first and second functional component within one of the container components;

the first functional component performs a second predetermined functionality and comprises a first functional coordinator for allowing the second functional component to communicate with the first functional component.

2. A system according to claim 1 wherein the first container component can be dynamically inserted into the first general runtime component, at which point the first container component introduces its first container coordinator into the first general runtime component.

3. A system according to claim 2 wherein the second container component may then establish a connection with the first container component via the first container coordinator.

4. A system according to claim 3 wherein the first functional component can be dynamically inserted into the first container component, at which point the first functional component can introduce its first functional coordinator into the first container component.

5. A system according to claim 4 wherein the second function component may then establish a connection with the first container component via the first container coordinator.

6. A system according to claim 5 wherein the first functional component may be dynamically terminated from the first container component.

7. A system according to claim 6 wherein the first functional component will enter a functional termination phase upon being dynamically terminated and will terminate the first functional coordinator prior to completing the functional termination phase.

8. A system according to claim 7 wherein the first container component may be dynamically terminated from the first general runtime component.

9. A system according to claim 8 wherein the first container component will enter a container termination phase upon being dynamically terminated and will terminate the first functional component prior to completing the container termination phase.

10. A method for implementing a modular software system on a distributed computing platform, the distributed computing platform having a plurality of connected processing resources, the method comprising:

providing a first and a second general runtime component which are capable of operating on one of the plurality of processing resources and of communicating with each other, and each of which comprise a plurality of sockets for

10488-15:1 4314314.1 PortInd2

connecting to a lower level component and managing interactions between the lower level components;

providing a first and a second container component for implementing a first and second predetermined functionality, respectively,

dynamically connecting the first and second container components into the one of the plurality of sockets in the first and second general runtime components, respectively;

with the first container component, inserting a first container coordinator into the first general runtime component, thereby creating a connection point for the second container component to initiate communication with the first container component by connecting to the first container coordinator;

providing a first and second functional component for performing a third and fourth predetermined functionality, respectively, within the first container component;

dynamically introducing the first functional component into the first container component; and

with the first functional component, inserting a first functional coordinator into the first container component, thereby creating a connection point for the second functional component to initiate communication with the first functional component by connecting to the first functional coordinator.

11. A method according to claim 10 further comprising:

dynamically removing the first container component from the first general runtime component;

in the first container component, terminating the first functional component.

12. A method according to claim 11 further comprising:

in the first functional component; terminating and withdrawing the first functional coordinator.